

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-265543

(43)Date of publication of application : 07.10.1997

(51)Int.Cl.

G06T 11/00
G06F 9/38

(21)Application number : 08-072464

(71)Applicant : MITSUBISHI ELECTRIC CORP

(22)Date of filing : 27.03.1996

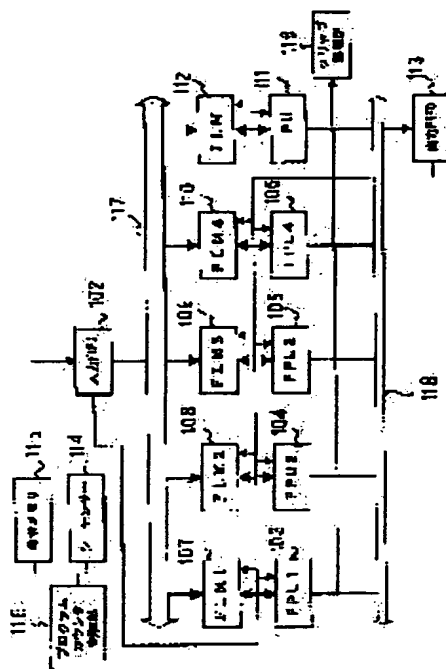
(72)Inventor : NEGISHI HIROYASU
KAMEYAMA MASATOSHI
INOUE YOSHITSUGU
KAWAI HIROYUKI

(54) GEOMETRY ARITHMETIC UNIT

(57)Abstract:

PROBLEM TO BE SOLVED: To increase the parallelism of parallel processing and enable fast geometry operation by providing floating-point arithmetic units which are connected to local memories connected to a global bus and operate independently of one another.

SOLUTION: The global bus 117 connects the local memories FLM107-FLM110 to an input FIFO 102. Data of the FIFO 102 are transferred by a sequencer 114 to the FLMs 107-110 according to an instruction in an instruction memory 115, as required. When the sequencer 114 instructs the arithmetic processings of the floating-point arithmetic units FPU103-FPU106 according to an instruction in the instruction memory 115, data need for the operation are read out of the FLMs 107-110 connected to the respective FPUs 103-106, which operate them, respectively. The processings of the four FPUs 103-106 are specified with one instruction and the FPUs 103-106 performs the same operation.



LEGAL STATUS

[Date of request for examination] 24.11.1999

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3203180

[Date of registration] 22.06.2001

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-265543

(43) 公開日 平成9年(1997)10月7日

(51) Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 T 11/00			G 0 6 F 15/72	3 8 0
G 0 6 F 9/38	3 7 0		9/38	3 7 0 C
			15/72	A

審査請求 未請求 請求項の数9 O L (全 23 頁)

(21) 出願番号 特願平8-72464

(22) 出願日 平成8年(1996)3月27日

(71) 出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72) 発明者 根岸 博康

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

(72) 発明者 亀山 正俊

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

(72) 発明者 井上 喜嗣

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

(74) 代理人 弁理士 宮田 金雄 (外3名)

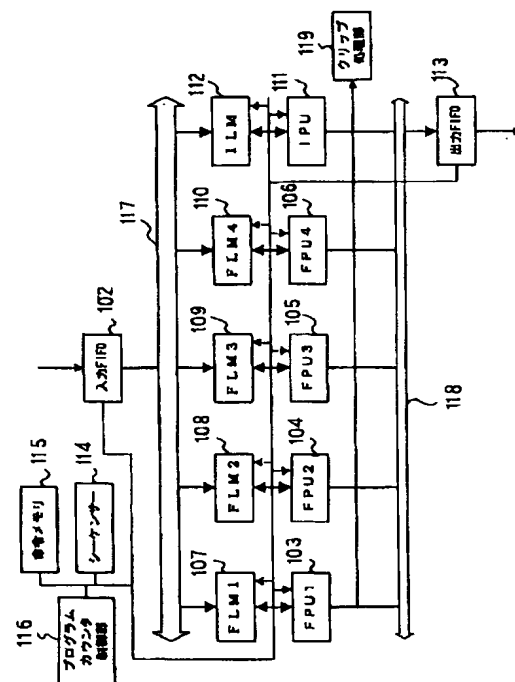
最終頁に続く

(54) 【発明の名称】 幾何学演算装置

(57) 【要約】

【課題】 3次元グラフィックスにおける幾何学演算処理を高速におこなうことを目的とする。

【解決手段】 グラフィックスの座標変換、輝度計算又はクリップ処理をはじめとする幾何学演算に必要なデータを記憶する入力メモリ102と、入力メモリ102を接続するグローバルバス117と、グローバルバス117に接続され、幾何学演算に必要なデータが転送される複数の浮動小数点演算用メモリ107~110と、入力メモリ102に記憶された幾何学演算に必要なデータを複数の浮動小数点演算用メモリ107~110に転送するシーケンサ114と、複数の浮動小数点演算用メモリ107~110と一対一に接続され、各浮動小数点演算用メモリ107~110に転送されたデータを用いて幾何学演算をそれぞれ独立に実行する複数の浮動小数点演算ユニット103~106とを備えたものである。



【特許請求の範囲】

【請求項1】 グラフィックスの座標変換、輝度計算又はクリップ処理をはじめとする幾何学演算に必要なデータを記憶する入力メモリと、
該入力メモリを接続するグローバルバスと、
該グローバルバスに接続され、前記幾何学演算に必要なデータが転送される複数の浮動小数点演算用メモリと、
前記入力メモリに記憶された前記幾何学演算に必要なデータを前記複数の浮動小数点演算用メモリに転送するシーケンサと、
前記複数の浮動小数点演算用メモリと一対一に接続され、前記各浮動小数点演算用メモリに転送されたデータを用いて前記幾何学演算をそれぞれ独立に実行する複数の浮動小数点演算ユニットとを備えたことを特徴とする幾何学演算装置。

【請求項2】 前記複数の浮動小数点演算ユニットと一対一に接続され、前記複数の浮動小数点演算ユニットの前記クリップ処理により生成されたクリップコードを記憶するとともに、該クリップコードをシフトする複数のシフトレジスタと、
該複数のシフトレジスタに記憶された前記クリップコードを用いて、複数の頂点で構成されるオブジェクトの状態を示すクリップ状態コードを生成するクリップ状態コード生成器と、
前記複数のシフトレジスタによりシフトされた前記クリップコードと前記クリップ状態コード生成器により生成された前記クリップ状態コードとを記憶するクリップコードレジスタとを備えたことを特徴とする請求項1記載の幾何学演算装置。

【請求項3】 前記シーケンサは、
前記複数の浮動小数点演算用メモリ間でのデータ転送の転送元を示すソースフィールドと、
前記データ転送の転送先を示すデスティネーションフィールドと、
前記ソースフィールドと前記デスティネーションフィールドとに設けられ前記複数の浮動小数点演算用メモリのいずれか又は全部を指定するビットフィールドとを有する転送命令により、データ転送を行うことを特徴とする請求項1記載の幾何学演算装置。

【請求項4】 幾何学演算の分岐処理を実行する特定の条件を記憶するターゲットレジスタと、
前記分岐処理を実行する複数の条件をそれぞれ記憶する複数のリファレンスレジスタと、
該リファレンスレジスタの各条件に一致する場合の処理を行う処理プログラムの格納アドレスを記憶するアドレステーブルと、
前記ターゲットレジスタと前記複数のリファレンスレジスタとを比較し、結果を出力する比較器と、
該比較器の比較結果に基づいて、前記アドレステーブルに記憶された前記アドレスの処理プログラムを実行する

プログラムカウンタ制御部とを備えたことを特徴とする幾何学演算装置。

【請求項5】 前記ターゲットレジスタと前記リファレンスレジスタとに設定された条件にマスクをかけるためのマスク情報を記憶するマスクレジスタを備えたことを特徴とする請求項4記載の幾何学演算装置。

【請求項6】 前記比較器の比較結果をエンコードして前記処理プログラムを格納した場所を示すポインタを生成するエンコーダと、

10 該エンコーダにより生成されたポインタに対するオフセットを記憶するオフセットレジスタと、
前記エンコーダにより生成されたポインタと前記オフセットレジスタに記憶されたオフセットとを加算する加算器とを備えたことを特徴とする請求項4記載の幾何学演算装置。

【請求項7】 幾何学演算に必要なデータを記憶する複数の浮動小数点演算用メモリと、

20 該複数の浮動小数点演算用メモリと一対一に接続され、
該各浮動小数点演算用メモリに記憶されたデータを用いて前記幾何学演算をそれぞれ独立に実行する複数の浮動小数点演算ユニットと、
該複数の浮動小数点演算ユニットと一対一に接続され、
該複数の浮動小数点演算ユニットのクリップ処理により生成されたクリップコードを記憶するとともに、該クリップコードをシフトする複数のシフトレジスタと、
該複数のシフトレジスタに記憶された前記クリップコードを用いてクリップ状態コードを生成するクリップ状態コード生成器と、
前記複数のシフトレジスタによりシフトされた前記クリップコードと前記クリップ状態コード生成器により生成された前記クリップ状態コードとを記憶するクリップコードレジスタとを備えたことを特徴とする請求項6記載の幾何学演算装置。

【請求項8】 コマンドを識別するフィールドと該コマンドに対応する処理プログラムを格納したアドレスが設定されるフィールドとを有するコマンドデータを記憶するコマンドレジスタを備え、

前記プログラムカウンタ制御部は、前記コマンドレジスタに設定されたアドレスの処理プログラムを実行することを特徴とする請求項4記載の幾何学演算装置。

【請求項9】 前記コマンドデータは、前記ターゲットレジスタに設定するための分岐条件が設定されるフィールドを有することを特徴とする請求項8記載の幾何学演算装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、3次元コンピュータグラフィックス表示システムにおいて、座標変換や輝度計算やクリップ処理などの幾何学処理演算を高速におこなうための装置に関するものである。

【0002】

【従来の技術】図24は従来例1とする特開昭63-86079号公報で示された従来の並列プロセッサ構成の3次元画像生成処理装置であり、3次元グラフィックス画像生成に大量に含まれる3次元ベクトル演算や行列演算を並列に実行することにより高速におこなうものである。図24において、IFはシステムバス10とのインターフェイス部、DBM1~4は画像生成処理に使用する物体形状データや処理パラメータを格納するデータメモリ、FPU1~3は浮動小数点演算ユニット、FALUは算術論理演算ユニット、20はデータメモリDBM1~4と浮動小数点演算ユニットFPU1~3と算術論理演算ユニットFALUとを結合する高速データバス、DCは演算器で作成した画像データを収集してディスプレイメモリに書き込むデータコレクタ、WCSは各演算ユニットやメモリ、論理回路を制御するための命令を格納するプログラムメモリ、SEQはプログラムメモリWCSに格納された命令を取り出すシーケンサ、ADGはシーケンサSEQによって取り出された命令のうち、メモリアドレス指定命令からデータメモリDBMの物理アドレスを生成するアドレス生成器である。

【0003】図25は図24の浮動小数点演算ユニットFPU1~3と算術論理演算ユニットFALUの組み合わせの具体例を詳細に示したものである。図25のFPP1~4は浮動小数点演算器、ALUは算術論理演算器、MPXは多数のデ*

$$\begin{pmatrix} X' \\ Y' \\ Z' \\ W' \end{pmatrix} = \begin{pmatrix} T11 & T12 & T13 & T14 \\ T21 & T22 & T23 & T24 \\ T31 & T32 & T33 & T34 \\ T41 & T42 & T43 & T44 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix} \quad (1)$$

【0006】図27は従来例2とする特開昭64-64034号公報に示されたシーケンス制御装置のブロック図である。この従来例2はシーケンス回路ハードウェアにより入力情報がある特定のパターンであるかどうかを判定可能にするものである。2701-1~2701-Nは入力されたN個の条件を示す2値の信号を保持するフリップフロップ、2702はフリップフロップ2701-1~2701-Nのデータを入力としてNビットのデータを生成する入力データ生成回路、2704-1は命令コードの入っている命令レジスタ、2703は入力データ生成回路2702の出力であるNビットの入力データと命令レジスタ2704-1の第2フィールド(2704-1a)にあたるNビットのデータを比較して一致したときに一致信号を出力する比較回路、2704-2は命令レジスタ2704-1の第1フィールドに保持されている条件分岐命令を解釈して実行信号を出力するデコーダ、2705は比較回路2703の出力する一致信号とデコーダ2704-2の出力する実行信号の論理積を出力する論理積回路、2706は論理積回路2705の出力信号を受けたときに条件分岐命令の分岐先を出力する分岐先選択回路、2704-3は分岐先選択回路2706の信号を受けて分岐先命令の分岐先アドレス

*ータから1つのデータを選択するマルチプレクサ、REG1~4は演算データを記憶するレジスタ、LUTは平方根や三角関数等の関数を高速に演算するためのパラメータを格納する参照テーブルである。図25に示すように各FPPの出力は他のFPPの入力になるようにマルチプレクサMPXを介してたすきがけ状に接続されている。このように4つの独立に動作する浮動小数点演算器FPP1~4を相互にたすきがけ状に接続することで、4つの独立した演算を並列におこなうことができる。

10 【0004】図26にこの並列プロセッサを利用した座標変換処理の行列演算のステップを示す。式(1)の行列成分であるT11, T12, T13, T14のデータは浮動小数点演算器FPP1に、T21, T22, T23, T24のデータは浮動小数点演算器FPP2に、T31, T32, T33, T34のデータは浮動小数点演算器FPP3に、T41, T42, T43, T44のデータは浮動小数点演算器FPP4に予め設定されているとすると、式1を計算するには16回の乗算と12回の加算が必要であるが、この例では、図26に示したように浮動小数点演算器FPP1, FPP2, FPP3, FPP4で1~12ステップの演算で実行可能である。しかし、現状ではさらなる高速化が望まれている。

【0005】

【数1】

を保持するプログラムカウンタである。

【0007】次にこのシーケンス回路の動作を説明する。Nビットデータを8ビットに仮定すると、8ビットの入力情報は第1~第8のフリップフロップ2701-1~2701-8に保持されており、入力データ生成回路2702で8ビットのデータを生成する。命令レジスタ2704-1の第2フィールドには“00000011”のデータが保持されているとする。比較回路2703はフリップフロップ2701-1~2701-8のデータが“00000011”の場合に一致信号を出力する。論理積回路2705はこの一致信号と命令レジスタ2704-1の第1フィールドにある条件分岐命令の命令コードをデコーダ2704-2が解釈して出力した実行信号との論理積を出力信号として出力する。この出力信号をもとに、分岐先選択回路2706は命令レジスタ2704-1の第3フィールドの分岐先アドレスを分岐先信号としてある処理の先頭アドレスを出力する。プログラムカウンタ2704-3は分岐先選択回路2706の出力するアドレスを保持して、システム全体としてはプログラムカウンタ2704-3の番地から実行を開始する。

【0008】図28は従来例3を説明する図であり、特

開平3-1264号公報で示されるマルチプロセッサ構成におけるマルチキャスト転送に関するものである。2801はメインプロセッサ、2802はI/Oプロセッサですべて独立して動作する。このような構成において、あるI/Oプロセッサ2802がバス上に宛先を示す情報であるアドレスデータを流すと、全てのI/Oプロセッサ2802は独自にバスに流れているアドレスデータを読み込み自分のプロセッサ宛のデータかどうかを判定する手段を持ち各I/Oプロセッサ2802が自分のデータかどうかを判定し、自分宛ての場合はデータを読み込んでいた。このような方式において、複数のI/Oプロセッサ宛てのアドレスデータを流すことにより、1つの転送元から複数の転送先に転送をおこなうマルチキャスト転送をおこなっていた。

【0009】また、従来のクリッピング処理はクリップコード発生回路において、X,Y,Z軸方向の面に対してクリップコード生成を逐次おこなって、ポリゴンのクリップ判定をおこなっていた。このため、クリップ判定に時間がかかり高速な描画が出来なかった。

【0010】

【発明が解決しようとする課題】従来のシステムは以上のように構成されているので、座標変換処理などでは並列プロセッサにより4つ同時に動作可能にもかかわらず、図26に示すように各プロセッサには動作しないステップがあり、処理時間がかかるという問題点があった。大量な計算をリアルタイムにおこなう3次元グラフィックス処理においては、並列処理の並列度を高めて処理時間を削減する必要がある。

【0011】また、従来の分岐シーケンスでは分岐先を一つ決めて、パターンがこの分岐条件に一致した場合に分岐するという処理しかできない。分岐先とパターンが複数ある場合には1つずつ分岐先ごとに比較する必要があるが高速な分岐処理が出来ないという問題点があった。

【0012】さらに、従来の並列プロセッサでは各プロセッサが自分のデータかどうかを判定する必要があるため回路が複雑になる。また従来のSIMD形式の並列プロセッサ構成においては、一つのプロセッサから他のプロセッサ全部に対してデータを転送する場合には、デスティネーションとなるプロセッサをひとつずつ指定していく必要があり転送時間がかかるという問題点があった。

【0013】また、従来のクリップ判定処理では、X,Y,Z座標に対して逐次処理する必要がある、クリップ判定処理に時間がかかるという問題点があった。

【0014】この発明は上記のような問題点を解消するためになされたもので、並列処理の並列度をあげてプロセッサの無駄な時間を省き、3次元グラフィックスの座標変換やベクトル計算を高速におこなうことを目的としている。また、条件分岐の際に複数の分岐条件を同時判定することで高速な条件分岐をおこなうことを目的としている。さらに、複雑な回路を付加することなく転送元から複数の指定した転送先へ同時に転送するマルチキャ

スト転送をおこなうことを目的としている。また、クリップ判定処理においてX,Y,Z,W座標に対して同時に処理をおこなうことで高速なクリップ判定処理をおこなうことを目的としている。

【0015】

【課題を解決するための手段】第1の発明は、グラフィックスの座標変換、輝度計算又はクリップ処理をはじめとする幾何学演算に必要なデータを記憶する入力メモリと、該入力メモリを接続するグローバルバスと、該グローバルバスに接続され、前記幾何学演算に必要なデータが転送される複数の浮動小数点演算用メモリと、前記入力メモリに記憶された前記幾何学演算に必要なデータを前記複数の浮動小数点演算用メモリに転送するシーケンサと、前記複数の浮動小数点演算用メモリと一対一に接続され、前記各浮動小数点演算用メモリに転送されたデータを用いて前記幾何学演算をそれぞれ独立に実行する複数の浮動小数点演算ユニットとを備えたものである。

【0016】第2の発明は、前記複数の浮動小数点演算ユニットと一対一に接続され、前記複数の浮動小数点演算ユニットの前記クリップ処理により生成されたクリップコードを記憶するとともに、該クリップコードをシフトする複数のシフトレジスタと、該複数のシフトレジスタに記憶された前記クリップコードを用いて、複数の頂点で構成されるオブジェクトの状態を示すクリップ状態コードを生成するクリップ状態コード生成器と、前記複数のシフトレジスタによりシフトされた前記クリップコードと前記クリップ状態コード生成器により生成された前記クリップ状態コードとを記憶するクリップコードレジスタとを備えたものである。

【0017】第3の発明は、前記複数の浮動小数点演算用メモリ間でのデータ転送の転送元を示すソースフィールドと、前記データ転送の転送先を示すデスティネーションフィールドと、前記ソースフィールドと前記デスティネーションフィールドとに設けられ前記複数の浮動小数点演算用メモリのいずれか又は全部を指定するビットフィールドとを有する転送命令により、データ転送を行うシーケンサを備えたものである。

【0018】第4の発明は、幾何学演算の分岐処理を実行する特定の条件を記憶するターゲットレジスタと、前記分岐処理を実行する複数の条件をそれぞれ記憶する複数のリファレンスレジスタと、該リファレンスレジスタの各条件に一致する場合の処理を行う処理プログラムの格納アドレスを記憶するアドレステーブルと、前記ターゲットレジスタと前記複数のリファレンスレジスタとを比較し、結果を出力する比較器と、該比較器の比較結果に基づいて、前記アドレステーブルに記憶された前記アドレスの処理プログラムを実行するプログラムカウンタ制御部とを備えたものである。

【0019】第5の発明は、前記ターゲットレジスタと前記リファレンスレジスタとに設定された条件にマスク

をかけるためのマスク情報を記憶するマスクレジスタを備えたものである。

【0020】第6の発明は、前記比較器の比較結果をエンコードして前記処理プログラムを格納した場所を示すポインタを生成するエンコーダと、該エンコーダにより生成されたポインタに対するオフセットを記憶するオフセットレジスタと、前記エンコーダにより生成されたポインタと前記オフセットレジスタに記憶されたオフセットとを加算する加算器とを備えたものである。

【0021】第7の発明は、幾何学演算に必要なデータを記憶する複数の浮動小数点演算用メモリと、該複数の浮動小数点演算用メモリと一対一に接続され、該各浮動小数点演算用メモリに記憶されたデータを用いて前記幾何学演算をそれぞれ独立に実行する複数の浮動小数点演算ユニットと、該複数の浮動小数点演算ユニットと一対一に接続され、該複数の浮動小数点演算ユニットのクリップ処理により生成されたクリップコードを記憶するとともに、該クリップコードをシフトする複数のシフトレジスタと、該複数のシフトレジスタに記憶された前記クリップコードを用いてクリップ状態コードを生成するクリップ状態コード生成器と、前記複数のシフトレジスタによりシフトされた前記クリップコードと前記クリップ状態コード生成器により生成された前記クリップ状態コードとを記憶するクリップコードレジスタとを備えたものである。

【0022】第8の発明は、コマンドを識別するフィールドと該コマンドに対応する処理プログラムを格納したアドレスが設定されるフィールドとを有するコマンドデータを記憶するコマンドレジスタを備え、前記プログラムカウンタ制御部は、前記コマンドレジスタに設定されたアドレスの処理プログラムを実行するものである。

【0023】第9の発明は、前記ターゲットレジスタに設定するための分岐条件が設定されるフィールドを有するコマンドデータをコマンドレジスタに記憶するものである。

【0024】

【発明の実施の形態】

実施の形態1. 以下に、この発明の一実施の形態を説明する。図1はこの実施の形態1の幾何学演算装置の構成を示すブロック図である。この実施の形態1では、4つの浮動小数点演算ユニットとそれに接続される4つのローカルメモリ、1つの整数型演算ユニットとそれに接続される1つのローカルメモリにより構成される形態を説明する。

【0025】図1において、102は図1には記述していないホストプロセッサが出力したデータを保持する入力FIFO、103、104、105、106は座標変換や輝度計算などを使用する積和演算などを実行する浮動小数点演算ユニットのFPU1、FPU2、FPU3、FPU4である。107、108、109、110はFPU1-103、FPU2-104、FPU3-105、FPU4-106のそれぞれに接続さ

れて演算に必要なパラメータとデータを保持する浮動小数点演算用メモリとしてのローカルメモリのFLM1、FLM2、FLM3、FLM4である。111は汎用的な整数演算をおこなう整数型演算ユニットのIPU、112は整数型演算ユニット111に接続されて演算に必要なパラメータとデータを保持するローカルメモリであるILM、113はFPU0-103、FPU1-104、FPU2-105、FPU3-106、IPU-111の出力データを保持する出力FIFOである。

【0026】114は命令メモリ115に保持される動作命令プログラムに基づきFPU1-104、FPU2-105、FPU3-106、IPU-111の演算制御と、FLM1-107、FLM2-108、FLM3-109、FLM4-110、ILM-112のメモリのリード/ライトの制御と、入力FIFO102のリード制御と、出力FIFO113のリード/ライト制御と、命令メモリ115から読み出す動作命令プログラム番地を生成し保持するプログラムカウンタ制御部116の制御と、クリップ処理部119の制御をおこなうシーケンサである。115はシーケンサ114の動作を指定する動作命令プログラムを保持する命令メモリ、116は分岐先プログラムを示すプログラムアドレスをプログラムカウンタへ設定するプログラムカウンタ制御部、117はローカルメモリFLM1~4及びILMと入力FIFO102を結ぶグローバルバス、118はFPU1-103、FPU2-104、FPU3-105、FPU4-106、IPU-111と出力FIFO-113を結ぶ出力バス、119はCohen-Sutherland法に基づくクリップ処理に使用するクリップフラグ生成をおこなうクリップ処理部である。

【0027】次に図1を用いて具体的な動作を説明する。3次元グラフィックス画像を生成するためには、描画オブジェクトの座標データと法線データと色属性データとオブジェクトを照らす光源データを基に計算処理をおこない、描画オブジェクトのディスプレイ上の座標と色データを生成する必要がある。ホストから上記座標、法線、色、光源データなどのデータを図1の幾何学演算装置に転送する際には入力FIFO102を介して行われ、入力FIFO102のデータは命令メモリ115の命令に従ってシーケンサ114が必要に応じてローカルメモリFLM1-107、FLM2-108、FLM3-109、FLM4-110、ILM-112のいずれかまたは複数の転送する。シーケンサ114が命令メモリ115内の命令に従ってFPU1-103、FPU2-104、FPU3-105、FPU4-106の演算処理を指示する場合は、演算に必要なデータをFPU1-103、FPU2-104、FPU3-105、FPU4-106のそれぞれに接続されているローカルメモリFLM1-107、FLM2-108、FLM3-109、FLM4-110から読みだし、各FPUが演算処理をおこなう。4つのFPUの処理は1つの命令で指定され、4つのFPUが同一の動作を行う。ある同一な処理を異なるデータで複数回実行する場合には、複数のFPUを結合して処理するのではなく一つ一つのFPUが単独で処理できるように構成しており、即ち、図25のようにたすきがけ状に結合しておらず、1回の命令で4つの同一動作を処理できるので、全体として1/4程度処理時間が短縮でき、4つの動作を指定する命令メモリ内のプログラムも1/4の量

で済むというメリットがある。

【0028】次に具体例をあげて動作を説明する。3次元処理の座標変換では(X, Y, Z, W)の同次座標系を用いることが多く、一例としてX座標はFPU1-103で、Y座標はFPU2-104で、Z座標はFPU3-105で、W座標はFPU4-106で処理するとする。座標変換処理はX, Y, Z, W座標それぞれに*

$$\begin{pmatrix} X' \\ Y' \\ Z' \\ W' \end{pmatrix} = \begin{pmatrix} M_{11} & M_{12} & M_{13} & M_{14} \\ M_{21} & M_{22} & M_{23} & M_{24} \\ M_{31} & M_{32} & M_{33} & M_{34} \\ M_{41} & M_{42} & M_{43} & M_{44} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix} \quad (2)$$

【0030】図2は実施の形態1の幾何学演算装置における座標変換処理のステップチャートである。あらかじめFLM1-107にX, Y, Z, WのデータとM11, M12, M13, M14のデータをロードし、FLM2-108にX, Y, Z, WのデータとM21, M22, M23, M24のデータをロードし、FLM3-109にX, Y, Z, WのデータとM31, M32, M33, M34のデータをロードし、FLM4-110にX, Y, Z, WのデータとM41, M42, M43, M44のデータをロードしておく。FPU1-103, FPU2-104, FPU3-105, FPU4-106に対して積和演算命令を指示することで、図2のように4ステップで4×4と1×4のマトリックス演算が可能になる。

【0031】また、輝度計算においては(R, G, B)の色データを用いるのが一般的であり、座標変換と同様に異なるデータに対する同一の処理で算出できる。R値を求めるのに必要なデータをFLM1-107に、G値を求めるのに必要なデータをFLM2-108に、B値を求めるのに必要なデータをFLM3-109に設定して計算をおこなうことができる。これにより、X, Y, Z, Wを1座標ごとにおこなうよりも1/4の時間で、R, G, Bを1つつづつ処理するよりも1/3の時間で処理を完了することができる。また従来例に比較してプロセッサが動作していない時間をなくしているため、従来例では12ステップかかる処理が4ステップで処理可能になる。

【0032】以上のように、この実施の形態によれば、グローバルバス117に接続されたローカルメモリFLM1~4と、このローカルメモリFLM1~4にそれぞれ接続され、各々が独立に演算を行う浮動小数点演算ユニットFPU1~4とを備えたことにより、並列処理の並列度が上がり、高速な幾何学演算を行うことができる。

【0033】なお、この実施の形態では、浮動小数点演算ユニットを4個備えた形態を示したが、4個でなくても良い。例えば、浮動小数点演算ユニットが2個の場合は、図2のFPU1, FPU2の処理を同時におこない、処理終了後にFPU3, FPU4の処理をおこなうので、8ステップで1頂点の処理ができる。また、3個の場合は2個の場合と同様に8ステップで処理でき、4個から7個の場合は4個の場合と同じで1頂点を4ステップで処理できる。8個から11個の場合はFPU1からFPU4までの1まとめ

*異なるデータで同一の処理を施せば処理が可能であり、変換は式(2)で表される。式(2)のM11~M44で示される行列は3次元空間での回転、拡大縮小、平行移動を示す行列である。

【0029】

【数2】

$$\begin{pmatrix} X' \\ Y' \\ Z' \\ W' \end{pmatrix} = \begin{pmatrix} M_{11} & M_{12} & M_{13} & M_{14} \\ M_{21} & M_{22} & M_{23} & M_{24} \\ M_{31} & M_{32} & M_{33} & M_{34} \\ M_{41} & M_{42} & M_{43} & M_{44} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix} \quad (2)$$

の処理を2組同時に実行させることができるので、2頂点を4ステップで処理できる。12個から15個の場合は3頂点を4ステップで処理できる。以上のように浮動小数点演算ユニットは4つでなくても良いが、4の倍数個の浮動小数点演算ユニットを使うのが望ましい。

【0034】実施の形態2. この発明の実施の形態2について説明する。図3は、実施の形態2の幾何学演算装置のブロック図である。図3において、103は実施の形態1のFPU1、104は実施の形態1のFPU2、105は実施の形態1のFPU3、304, 305, 306はFPU1-103, FPU2-104, FPU3-105にそれぞれ接続されてFPU1~3で生成される3頂点のクリップコードを保持する6ビットのシフトレジスタ、307はシフトレジスタ304, 305, 306のクリップコードを入力として3頂点で構成されるオブジェクトがクリップ領域を指定するクリップ枠とどのような関係にあるのかを示すクリップ状態コードを生成するクリップ状態コード生成器、308はクリップコードとクリップ状態コードを保持するクリップコードレジスタである。

【0035】クリップコードは図4に示すような6面(X=XMAX, X=XMIN, Y=YMAX, Y=YMIN, Z=ZMAX, Z=ZMIN)のクリップ枠で指定される直方体のクリップ領域の内部にあるか外部にあるかを示すもので、クリップ枠のX=XMAX, X=XMIN, Y=YMAX, Y=YMIN, Z=ZMAX, Z=ZMINのそれぞれに対して頂点がクリップ領域の内部にある場合には0、外部にある場合には1を割り当てる。図5にクリップコードを示す。図5は図4の直方体のクリップ領域をZ=ZMAX, Z=ZMINによってZがZMAXより大きい場合の領域(図5(a))、ZがZMAXとZMINの間の場合の領域(図5(b))、ZがZMINより小さい場合の領域(図5(c))に分けて示したもので図4の原点からZ正方向に見たときの図である。また、図5(d)に示すように、クリップコードの0ビット目はX軸方向の最大値(XMAX)を持つクリップ枠に対するコード、1ビット目はX軸方向の最小値(XMIN)を持つクリップ枠に対するコード、2ビット目はY軸方向の最大値(YMAX)を持つクリップ枠に対するコード、3ビット目はY軸方向の最小値(YMIN)を持つクリップ枠に対するコード、4ビット目はZ軸方向の最大値(ZMAX)を持つクリップ枠に対するコード、5ビット

目はZ軸方向の最小値(ZMIN)を持つクリップ枠に対するコードを割り当てることで、3次元の立方体のクリップ領域に対するクリップコードを生成できる。

【0036】図5(a)を例に取って具体的に説明すると、図5(a)はZ=ZMAXのクリップ枠よりもクリップ領域の外側なのでクリップコードの4ビット目は全て1になり、Z=MINのクリップ枠よりもクリップ領域の内側なので、クリップコードの5ビット目は全て0になる。X=XMAXよりもクリップ領域内部にあたる領域11, 12, 14, 15, 17, 18ではクリップコードの0ビット目は0になり、外側にあたる領域13, 16, 19はクリップコードの0ビット目は1になる。X=XMINよりもクリップ領域内部にあたる領域12, 13, 15, 16, 18, 19ではクリップコードの1ビット目は0になり、外側にあたる領域11, 14, 17はクリップコードの1ビット目は1になる。Y=YMAXよりもクリップ領域内部にあたる領域14, 15, 16, 17, 18, 19ではクリップコードの2ビット目は0になり、外側にあたる領域11, 12, 13はクリップコードの2ビット目は1になる。Y=YMINよりもクリップ領域内部にあたる領域11, 12, 13, 14, 15, 16ではクリップコードの3ビット目は0になり、外側にあたる領域17, 18, 19はクリップコードの3ビット目は1になる。

【0037】次に動作の概略を説明する。FPU1-103, FPU2-104, FPU3-105で生成されたクリップコードはシフトレジスタ304, 305, 306に入れられる。3頂点分のクリップコードを保持できるシフトレジスタ304, 305, 306は一番古いクリップコードを捨てるためにクリップコードをシフトする。そして、シフトレジスタ304, 305, 306のデータはクリップコードレジスタ308とクリップ状態コード生成器307に入力される。クリップ状態コード生成器307はシフトレジスタ304, 305, 306から入力されるクリップコードの論理演算によってクリップ状態コードを生成する。クリップ状態コードには描画オブジェクトがクリップ領域に入っているか(クリップイン)、入っていないか(クリップアウト)、交わっているか(クリップ)、描画オブジェクトがクリップ領域のXMAXと交わっているか、XMINと交わっているか、YMAXと交わっているか、YMINと交わっているか、ZMAXと交わっているか、ZMINと交わっているかを示すフラグがあり、これらクリップ状態コードはクリップレジスタ308に出力される。

【0038】次に具体的な例を示す。図6のようなクリップ領域と3角形オブジェクト(頂点1はY=YMAXに対してクリップ領域外、頂点2はX=XMAXに対してクリップ領域外、頂点3はクリップ領域内)があり、FPU1-103でX、FPU2-104でY、FPU3-105でZ方向のクリップ面に対するクリップコードを生成するとする。

【0039】頂点1のクリップコードを生成する命令を実行すると、FPU1-103ではクリップ面と頂点のX座標とで演算が行われてXMAX、XMIN面に対するクリップコード(XMAX, XMIN) 00が、FPU2-104ではクリップ面と頂点

Y座標とで演算がおこなわれてYMAX, YMIN面に対するクリップコード(YMAX, YMIN) 10が、FPU3-105ではクリップ面と頂点Z座標とで演算がおこなわれてZMAX, ZMIN面に対するクリップコード(ZMAX, ZMIN) 00が並列処理されることにより同時に生成され、頂点1のクリップコード(XMAX, XMIN, YMAX, YMIN, ZMAX, ZMIN) 001000が生成される。これらのX, Y, Zに関する3つのコードはFPU1-103, FPU2-104, FPU3-105それぞれに接続されているシフトレジスタ304, 305, 306の第1フィールドにロードされる。シフトレジスタ304, 305, 306の3頂点のクリップコードはクリップコードレジスタ308に入力されると共に、クリップ状態コード生成器307にも入力され、クリップ状態コードを生成する。この時点でのシフトレジスタ304, 305, 306のクリップコード、クリップ状態コードは図7のステップ1のように、それぞれのシフトレジスタ304, 305, 306の第1フィールドを取り出すと001000となるクリップコードが保持されている。

【0040】次に頂点2のクリップコードをFPU1-103で10、FPU2-104で00、FPU3-105で00を同様に生成し、それぞれのシフトレジスタ304, 305, 306をシフト後にロードし、クリップ状態コード生成器308でクリップ状態コードを生成する。この時点でのシフトレジスタ304, 305, 306とクリップ状態コードは図7のステップ2のようになっており、それぞれのシフトレジスタの第1フィールドのデータを取り出すと100000となる頂点2のクリップコードが保持され、第2フィールドのデータを取り出すと001000となる頂点1のクリップコードが保持され、クリップ状態コードにより頂点1と頂点2の2頂点で構成される直線がクリップ状態であることを判定できる。

【0041】次に頂点3のクリップコードをFPU1-103で00、FPU2-104で00、FPU3-105で00を同様に生成し、シフト後にシフトレジスタ304, 305, 306にロードし、クリップ状態コード生成器308でクリップ状態コードが生成される。この時点でのシフトレジスタ304, 305, 306とクリップ状態コードは図7のステップ3のようになっており、それぞれのシフトレジスタ304, 305, 306の第1フィールドのデータを取り出すと000000となる頂点3のクリップコードが保持され、第2フィールドのデータを取り出すと100000となる頂点2のクリップコードが保持され、第3フィールドのデータを取り出すと001000となる頂点1のクリップコードが保持され、頂点1、頂点2、頂点3の3頂点で構成される三角形はクリップ状態であることが判定できる。

【0042】オブジェクトのクリップ状態コードに応じたクリップイン状態の場合の処理とクリップアウト状態の場合の頂点データに対する破棄処理とクリップ状態の場合のクリップ枠外の部分を切り取るクリップ処理に分岐する際に、3頂点のクリップコードとクリップ状態コードが入力されているクリップコードレジスタをI P U

が参照することでクリップ判定がおこなえる。

【0043】以上のように、この実施の形態によれば、従来おこなっていたX、Y、Z面に対して逐次にクリップテストをおこなう処理に比べてX、Y、Zをまとめて同時処理できるため、逐次処理をしていた場合よりも処理時間を1/3にする高速なクリップ判定が可能である。

【0044】実施の形態3。この発明の実施の形態3について説明する。図8は実施の形態1のプログラムカウンタ制御部116の詳細を示したブロック図である。図8において、801はある条件の有効無効をビットで示したnビットパターンを設定するターゲットレジスタ、802は前記条件の中で特定の条件パターンを入れるリファレンスレジスタ、803は前記ターゲットレジスタ801のデータとリファレンスレジスタ802のデータを等しいかどうか比較し、リファレンスレジスタ数のビットを比較結果データとして出力する比較器、804はリファレンスレジスタ802の条件に対応するプログラムを格納しているアドレスを保持するアドレステーブル、805は図1のプログラムカウンタ制御部116内にあり、図1の命令メモリ15内の次の命令アドレスを指し示すプログラムカウンタである。

【0045】まず本実施の形態の背景を説明する。ある処理に対してN個の分類条件があり、それぞれに特有な処理が割り当てられているとすると、N個の条件により2のN乗の処理に分岐するためにはN回の分岐判定処理が必要になる。一般に2のN乗個の処理があるが条件は全て均等に生じるものではなく、頻繁におきる条件は2のN乗個のうち数個程度の条件である場合が多い。このような場合は頻繁におきる条件の処理は減多におきない条件のための判定を含めてN回の条件判定することにより処理に時間がかかる。このため頻繁に発生する条件を高速に見つけ出して対応する処理に高速に分岐する必要があり、ある特定の条件判定とそれに対応する処理への分岐を高速に行うために本実施の形態は考えられた。

【0046】次に図8を用いて動作を説明する。N個の条件をNビットパターンの各1ビットで有効無効を示す条件の組み合わせの中で条件分岐を高速におこないたい条件のビットパターンをリファレンスレジスタ802に入れる。また現在設定されている条件を示すビットパターンをターゲットレジスタ801に入力する。分岐処理命令が図8には記述されていないシーケンサによって起動されると比較器803内の比較回路にターゲットレジスタ801とリファレンスレジスタ802のデータが入力される。比較回路は2つのデータを比較した結果一致する場合には1を、一致しない場合には0を出力する。比較器803の一致した場合の出力に対応するアドレステーブル804のデータを読みだしプログラムカウンタ805に設定することで分岐をおこなう。

【0047】次に具体例として図9を用いて説明すると、ターゲットレジスタ801、リファレンスレジスタ802

を8ビットレジスタと仮定し、ターゲットレジスタ801には10101111が、リファレンスレジスタ802の6番目にターゲットレジスタ801と同じ10101111が、それ以外は別の値が入力されているとする。また、アドレステーブル804にはリファレンスレジスタ802の条件ごとの分岐先のプログラムアドレスを設定しておく。分岐処理命令が起動され、ターゲットレジスタ801とリファレンスレジスタ802の値を比較器803に読み込み、比較を行った結果、一致するのはリファレンスレジスタ802の6番目なので、00000100というデータを比較器803は出力する。図9に示されるように1が示されているビットに対応したテーブルから0x200を読みだしプログラムカウンタ805に設定することで0x200番地からのプログラムに分岐することができる。

【0048】以上のように、この実施の形態によれば、複数の条件のなかで特定の条件に対応する処理への分岐を高速におこなえ、特定パターンも複数用意することが可能になる。

【0049】実施の形態4。この発明の実施の形態4について説明する。図10は実施の形態4のプログラムカウンタ制御部の構成を示すブロック図であり、実施の形態3の構成にマスクレジスタを追加したものである。図10において、1001はマスクレジスタである。次に動作について説明する。リファレンスレジスタ802とターゲットレジスタ801との比較をする際に、各レジスタのビットにマスクをかけることでNビットの条件のなかである条件の比較の有効/無効を選択でき、細かい条件分岐が可能になる。この実施の形態では、マスクレジスタ1001のデータが1の場合に比較を無効、0の場合に比較を有効にする。

【0050】具体例として図11を用いて説明すると、ターゲットレジスタ801、リファレンスレジスタ802を8ビットレジスタと仮定し、ターゲットレジスタには11111111、リファレンスレジスタの6番目にはターゲットレジスタの上4ビットが同じ11111101が、それ以外は別の値が入力され、マスクレジスタには00001111が設定されているとする。また、アドレステーブル804にはリファレンスレジスタ802の条件に対応する分岐先のプログラムアドレスを設定しておく。分岐処理命令が起動され、ターゲットレジスタ801とリファレンスレジスタ803の値を比較器803に読み込み、比較する際に、マスクレジスタ1001の値でマスクされていない部分のみの比較をおこなう。マスクレジスタ1001の値が00001111なので上4ビットのみの比較をすることになり、比較した結果一致するのはリファレンスレジスタの6番目なので、00000100というデータを出力する。図11に示されるように1が示されているビットに対応したアドレステーブル804から0x200を読みだしプログラムカウンタ805に設定し、0x2

00番地からのプログラムに分岐することができる。

【0051】以上のように、この実施の形態によれば、マスク機能を追加することにより、条件判定のマスク処理が可能になり、条件判定を無視する指定ができるようになり多彩な条件判定が可能になる。

【0052】実施の形態5. この発明の実施の形態5について説明する。図12は実施の形態5のプログラムカウンタ制御部の構成を示すブロック図であり、実施の形態3の構成にエンコーダと加算器とオフセットレジスタを追加したものである。図12において、1201は比較器803の比較結果をエンコードするエンコーダ、1202はエンコーダ1201の結果にオフセットする値を保持するオフセットレジスタ、1203はエンコーダ1201の値とオフセットレジスタ1202の値を加算する加算器である。

【0053】次に動作について説明する。基本的な動作は実施の形態3と同様であり、オフセットレジスタ1202にエンコーダ1201の出力データのオフセット値を設定しておき、ターゲットレジスタ801とリファレンスレジスタ802の値を比較器803で比較した結果をエンコーダ1201に入力する。加算器1203はエンコーダ1201の出力とオフセットレジスタ1202の値を読み出して加算して、アドレステーブル804のアドレスとして出力する。このアドレスでテーブルを参照し、テーブルから読み出したデータをプログラムカウンタ805に設定し分岐をおこなう。

【0054】具体例として図13を用いて説明する。ターゲットレジスタ801、リファレンスレジスタ802を8ビットレジスタと仮定し、ターゲットレジスタ801には10101101が、リファレンスレジスタ802の6番目にターゲットレジスタと同じ10101101が、それ以外は別の値が入力され、オフセットレジスタ1202には0x50が設定されているとする。また、アドレステーブル804にはリファレンスレジスタ802とオフセットレジスタ1202の組み合わせ条件に対応する分岐先のプログラムアドレスを設定しておく。分岐処理命令が起動され、ターゲットレジスタ801とリファレンスレジスタ802の値を比較器803に読み込み、比較し、その結果をエンコーダ1201に入力して6番目のレジスタであるのでエンコードの結果である6とオフセットレジスタ1202のデータ0x50を加算器1203に入力する。加算器の出力0x56をアドレスとしてアドレステーブル804を参照する。図13には示していないシーケンサはアドレステーブル804の0x56番地からジャンプ先アドレスである0x280を読みだしプログラムカウンタ805に設定して0x280番地からのプログラムに分岐することができる。

【0055】以上のように、この実施の形態によれば、オフセット機能を追加することにより、分岐条件以外の要因による条件設定ができるようになり、より多彩な分岐が可能になる。

【0056】実施の形態6. この発明の実施の形態6に

ついて説明する。図14は実施の形態6のプログラムカウンタ制御部の構成を示すブロック図であり、実施の形態4と実施の形態5の構成を合わせたものである。

【0057】次に動作について説明する。基本的な動作は実施の形態4、5を合わせたものであり、ターゲットレジスタ801とリファレンスレジスタ802の比較を比較器803でおこなう。この際にマスクレジスタ1001の値でビット毎の比較の有効無効を設定する。マスクデータは1でマスクを無効、0でマスクを有効にする。比較結果はエンコーダ1201に入力されエンコードをおこなう。オフセットレジスタ1202にはエンコーダ1201の出力データに対するオフセット値を設定しておき、加算器1203でオフセット値とデコード結果を加算して、その結果をアドレステーブル804のアドレスとして参照する。図14には記載されていないシーケンサによりアドレステーブルのデータがプログラムカウンタ805に設定されて分岐をおこなう。オフセットレジスタ1202とマスクレジスタ1001とを組み合わせる細かい分岐が可能になる。

【0058】具体例として図15を用いて説明すると、ターゲットレジスタ801、リファレンスレジスタ802を8ビットレジスタと仮定し、ターゲットレジスタ801には11111011が、リファレンスレジスタ802の7番目に上4ビットがターゲットレジスタと同じ11110000が、7番目以外は上位4ビットが別の値で保持され、マスクレジスタ1001には00001111、オフセットレジスタ1401には0x100が設定されているとする。また、アドレステーブルにはリファレンスレジスタ802とオフセットレジスタ1401を組み合わせ条件に対応した分岐先のプログラムアドレスを設定しておく。分岐処理命令が起動され、ターゲットレジスタ801とリファレンスレジスタ802の値を比較器803に読み込み、比較する際にマスクレジスタ1001の値でマスクされていない部分のみの比較をおこなう。マスクレジスタで上4ビットがマスクされていないので、一致するリファレンスレジスタ802の7番目であり、比較器803は00000010というデータを出力する。このデータはエンコーダに入力されエンコードの結果、7番目のデータのため出力するデータ7にオフセットレジスタ1202のデータ0x100が加算器1203で加算され、0x107を出力する。このアドレスでアドレステーブル804を参照して、シーケンサはアドレステーブル804の0x107番地からジャンプ先アドレスである0x340を読みだしプログラムカウンタ805に設定し、0x340番地からのプログラムを実行する。

【0059】以上のように、この実施の形態によれば、マスク機能とオフセット機能を追加することにより、条件判定のマスク処理と条件判定を無視する指定とができるようになったため、多彩な条件判定が可能になり、また分岐条件以外の要因による条件設定もできるようになり、より多彩な分岐が可能になる。

【0060】実施の形態7. この発明の実施の形態7について説明する。図16は、実施の形態7のプログラムカウンタ制御部の構成を示すブロック図である。図において、103はFPU1、104はFPU2、105はFPU3、304はシフトレジスタ1、305はシフトレジスタ2、306はシフトレジスタ3、307はクリップ状態コード生成器、308はクリップコードレジスタ、1201はエンコーダ、1202はオフセットレジスタ、1203は加算器、804はアドレステーブル、805はプログラムカウンタである。

【0061】次に動作について説明する。基本的な動作は実施の形態2と実施の形態5を組み合わせたもので、実施の形態5にクリップコードレジスタ308の内容による分岐ジャンプを可能にしたものである。例えば3角形をクリップ判定する場合にはFPU1-103、FPU2-104、FPU2-105で作成された3頂点のクリップコードをシフトレジスタ304、305、306を介して、クリップ状態コード生成器307で論理演算をおこなうことでクリップ状態コードを生成し、クリップコードレジスタ308に保持する。クリップコードレジスタ308に保持されているクリップ状態コードのクリッピング・クリップアウト・クリップの3つの状態ビットを実施の形態5と同様なエンコーダ1201に入力してエンコードをおこなう。エンコード結果とオフセットレジスタ1202の値を加算器1203で加算した結果でアドレステーブル804を参照し、参照したデータをプログラムカウンタ805に設定する。オフセットレジスタ1202には、オブジェクト（3角形、直線など）ごとのクリップ処理ルーチンを保持するアドレスを設定する。

【0062】以上のように、この実施の形態によれば、クリップコード判定とクリップ処理ルーチンへのジャンプをH/Wで処理することができるので、高速なクリップ判定が可能になる。

【0063】実施の形態8. この発明の実施の形態8について説明する。図17は実施の形態1で示した4つの浮動小数点演算ユニットと1つの整数型演算ユニットからなる場合の幾何学演算処理装置のローカルメモリのFLM1-107、FLM2-108、FLM3-109、FLM4-110、ILM-112と入力FIFO102を示したものである。ホストから入力され、入力FIFO102に保持されているデータを各FLM1~4に転送する場合に、図1のシーケンサ114がマルチキャスト転送命令を解釈し、1つの転送元から複数の転送先を指定して同時に転送するマルチキャスト転送の指示をしてマルチキャスト転送をおこなうことにより、1サイクルで4つのメモリに同時転送が可能になる。

【0064】このマルチキャスト転送を実現するために、転送命令の命令フィールドを図18のように定義した。コマンドフィールド、転送元のアドレスを示すソースアドレス、及び、転送先のアドレスを示すデスティネーションアドレスに加えて、転送元を示すソースフィールドと、転送先を示すデスティネーションフィールドとを追加した。この2つのフィールドには図19に示すよ

うな内部レジスタ、ILM-112、マルチキャストFLM、FLM1-107、FLM2-108、FLM3-109、FLM4-110等が設定できる。例えばソースフィールドにFLM1-107、デスティネーションフィールドにマルチキャストFLMを設定すると、FLM1-107からFLM1~4へのマルチキャスト転送が可能になる。

【0065】以上のように、この実施の形態によれば、マルチキャスト転送により、並列処理をおこなうプロセッサ間で効率のよいデータ転送をサポートできる。

【0066】実施の形態9. この発明の実施の形態9について説明する。図20は実施の形態3の幾何学演算装置のプログラムカウンタ制御部にコマンドレジスタ2001を加えたものである。ホストから入力される幾何学演算装置の制御コマンドにはコマンドを識別可能なコマンドIDが含まれているが、本実施の形態ではこれに加えてコマンドに対応するプログラムを保持する命令メモリのアドレスを含む。図21に示すようにコマンドデータの中にプログラムアドレスを入れるフィールドを追加してある。

【0067】図20で動作を説明すると、ホストから入力されたコマンドは一旦コマンドレジスタ2001に入り、実施の形態3で示した条件分岐をおこなうコマンドかどうかを調べる。実施の形態3で示した分岐を行わないコマンドの場合にはコマンドデータに含まれるプログラムアドレスを直接プログラムカウンタ805に設定することで、コマンドデータの処理プログラムにジャンプする。

【0068】以上のように、この実施の形態によれば、コマンドIDを解析してプログラムを実行するよりも高速な動作が可能になる。

【0069】実施の形態10. この発明の実施の形態10について説明する。図22は実施の形態3の幾何学演算装置のプログラムカウンタ制御部にコマンドレジスタ2001を加えたものである。ホストから入力されるコマンドにはコマンドを識別可能なコマンドIDが含まれているが、本実施の形態ではこれに加えて実施の形態3で示した条件分岐をおこなう際にターゲットレジスタ801に設定する分岐条件のデータを含む。図23に示すようにコマンドを示すデータの中に条件ビットフィールドを追加してある。

【0070】図22で動作を説明すると、ホストから入力されたコマンドは一旦コマンドレジスタ2001に入り、実施の形態3で示した分岐をおこなうコマンドかどうかを調べる。実施の形態3で示した分岐をおこなうコマンドの場合にはコマンドデータの条件ビットフィールドのデータがターゲットレジスタ801に入れられる。

【0071】以上のように、この実施の形態によれば、実施の形態3で示される分岐を必要とするコマンドを実行する前に、ターゲットレジスタ801に条件ビットを設定するコマンドを入力する必要がなくなり、分岐を高速におこなうことができる。

【0072】

【発明の効果】第1の発明によれば、グローバルバスに接続され、幾何学演算に必要なデータが転送される複数の浮動小数点演算用メモリと、入力メモリに記憶された幾何学演算に必要なデータを複数の浮動小数点演算用メモリに転送するシーケンサと、複数の浮動小数点演算用メモリと一対一に接続され、各浮動小数点演算用メモリに転送されたデータを用いて幾何学演算をそれぞれ独立に実行する複数の浮動小数点演算ユニットとを備えたことにより、並列処理の並列度があがるので、幾何学演算を高速に実行することができる。

【0073】第2の発明によれば、複数の浮動小数点演算ユニットと一対一に接続され、複数の浮動小数点演算ユニットのクリップ処理により生成されたクリップコードを記憶するとともに、該クリップコードをシフトする複数のシフトレジスタと、該複数のシフトレジスタに記憶されたクリップコードを用いてクリップ状態コードを生成するクリップ状態コード生成器と、複数のシフトレジスタによりシフトされたクリップコードとクリップ状態コード生成器により生成されたクリップ状態コードとを記憶するクリップコードレジスタとを備えたことにより、クリップコードの計算を並列におこなえるので、高速にクリップ判定を実行することができる。

【0074】第3の発明によれば、複数の浮動小数点演算メモリ間でのデータ転送の転送元を示すソースフィールドと、データ転送の転送先を示すデスティネーションフィールドと、ソースフィールドとデスティネーションフィールドとに設けられ複数の浮動小数点演算用メモリのいずれか又は全部を指定するビットフィールドとを有する転送命令により、データ転送を行うシーケンサを備えたことにより、並列処理をおこなうプロセッサ間で、高速なマルチキャスト転送が実行できる。

【0075】第4の発明によれば、幾何学演算の分岐処理を実行する特定の条件を記憶するターゲットレジスタと、分岐処理を実行する複数の条件をそれぞれ記憶する複数のリファレンスレジスタと、該リファレンスレジスタの各条件に一致する場合の処理を行う処理プログラムの格納アドレスを記憶するアドレステーブルと、ターゲットレジスタと複数のリファレンスレジスタとを比較し、結果を出力する比較器と、該比較器の比較結果に基づいて、アドレステーブルに記憶されたアドレスの処理プログラムを実行するプログラムカウンタ制御部とを備えたことにより、条件分岐の際に複数の分岐条件を同時判定するので、高速な条件分岐を行うことができる。

【0076】第5の発明によれば、前記ターゲットレジスタと前記リファレンスレジスタとに設定された条件にマスクをかけるためのマスク情報を記憶するマスクレジスタを備えたことにより、条件判定を無視する指定ができるようになり、多彩な条件判定が可能になる。

【0077】第6の発明によれば、比較器の比較結果をエンコードして前記処理プログラムを格納した場所を示

すポインタを生成するエンコーダと、該エンコーダにより生成されたポインタに対するオフセットを記憶するオフセットレジスタと、エンコーダにより生成されたポインタとオフセットレジスタに記憶されたオフセットとを加算する加算器とを備えたことにより、分岐条件以外の要因による条件設定ができるようになり、より多彩な分岐が可能になる。

【0078】第7の発明によれば、複数の浮動小数点演算メモリと一対一に接続され、各浮動小数点演算用メモリに記憶されたデータを用いて幾何学演算をそれぞれ独立に実行する複数の浮動小数点演算ユニットと、該複数の浮動小数点演算ユニットと一対一に接続され、該複数の浮動小数点演算ユニットのクリップ処理により生成されたクリップコードを記憶するとともに、該クリップコードをシフトする複数のシフトレジスタと、該複数のシフトレジスタに記憶された前記クリップコードを用いてクリップ状態コードを生成するクリップ状態コード生成器と、複数のシフトレジスタによりシフトされたクリップコードとクリップ状態コード生成器により生成されたクリップ状態コードとを記憶するクリップコードレジスタとを備えたことにより、高速なクリップ判定が可能になる。

【0079】第8の発明によれば、コマンドを識別するフィールドと該コマンドに対応する処理プログラムを格納したアドレスが設定されるフィールドとを有するコマンドデータを記憶するコマンドレジスタを備え、プログラムカウンタ制御部は、コマンドレジスタに設定されたアドレスの処理プログラムを実行することにより、コマンドの種類を判定することなくプログラムカウンタにアドレスを設定できるので、プログラム起動が高速化できる。

【0080】第9の発明によれば、コマンドデータに、前記ターゲットレジスタに設定するための分岐条件が設定されるフィールドを設けたことにより、条件ビットを設定する前処理を省略できるため、プログラム起動が高速化できる。

【図面の簡単な説明】

【図1】 実施の形態1の幾何学演算装置のブロック図である。

【図2】 実施の形態1の幾何学演算装置における座標変換処理のステップチャートである。

【図3】 実施の形態2の幾何学演算装置のブロック図である。

【図4】 クリップ領域を示す図である。

【図5】 クリップコードを示す図である。

【図6】 クリップ領域とオブジェクトの具体例を示す図である。

【図7】 図6の具体例におけるレジスタ内容を示す図である。

【図8】 実施の形態3のプログラムカウンタ制御部の

構成を示すブロック図である。

【図 9】 実施の形態 3 のプログラムカウンタ制御部の具体例を示す図である。

【図 10】 実施の形態 4 のプログラムカウンタ制御部の構成を示すブロック図である。

【図 11】 実施の形態 4 のプログラムカウンタ制御部の具体例を示す図である。

【図 12】 実施の形態 5 のプログラムカウンタ制御部の構成を示すブロック図である。

【図 13】 実施の形態 5 のプログラムカウンタ制御部の具体例を示す図である。

【図 14】 実施の形態 6 のプログラムカウンタ制御部の構成を示すブロック図である。

【図 15】 実施の形態 6 のプログラムカウンタ制御部の具体例を示す図である。

【図 16】 実施の形態 7 のプログラムカウンタ制御部の構成を示すブロック図である。

【図 17】 実施の形態 8 の幾何学演算装置を説明する図である。

【図 18】 実施の形態 8 で示される転送命令の命令フォーマットである。

【図 19】 実施の形態 8 で示されるフィールドコードの一例である。

【図 20】 実施の形態 9 のプログラムカウンタ制御部の構成を示すブロック図である。

【図 21】 実施の形態 9 で示されるコマンドフィールドの一例である。

【図 22】 実施の形態 10 のプログラムカウンタ制御*

* 部の構成を示すブロック図である。

【図 23】 実施の形態 10 で示されるコマンドフィールドの一例である。

【図 24】 従来の 3 次元画像生成処理装置のブロック図である。

【図 25】 従来の 3 次元画像生成処理装置の詳細ブロック図である。

【図 26】 従来の 3 次元画像生成処理装置における座標変換処理のステップチャートである。

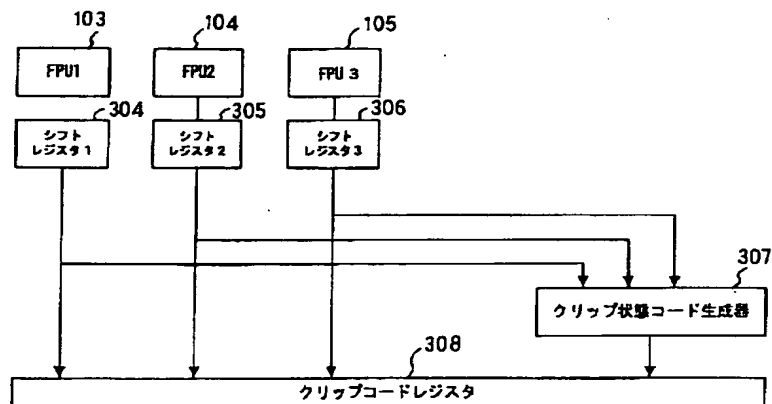
【図 27】 従来のシーケンス制御装置のブロック図である。

【図 28】 従来のマルチキャスト転送を説明する図である。

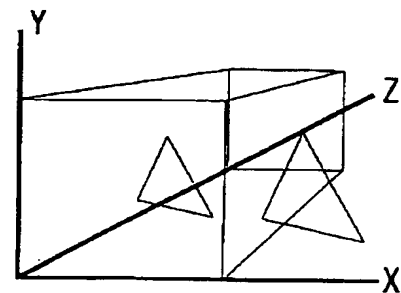
【符号の説明】

102 入力FIFO、103 FPU1、104 FPU2、105 FPU3、106 FPU4、107 FLM1、108 FLM2、109 FLM3、110 FLM4、111 IPU、112 ILM、113 出力FIFO、114シーケンサー、115 命令メモリ、116 プログラムカウンタ制御部、117 グローバルバス、118 出力バス、119 クリップ処理部、304 シフトレジスタ1、305 シフトレジスタ2、306 シフトレジスタ3、307 クリップ状態コード生成器、308 クリップコードレジスタ、801 ターゲットレジスタ、802 リファレンスレジスタ、803 比較器、804 アドレステーブル、805 プログラムカウンタ、1001 マスクレジスタ、1201 エンコーダ、1202 オフセットレジスタ、1203 加算器、2001 コマンドレジスタ

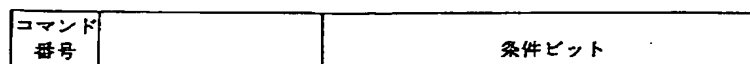
【図 3】



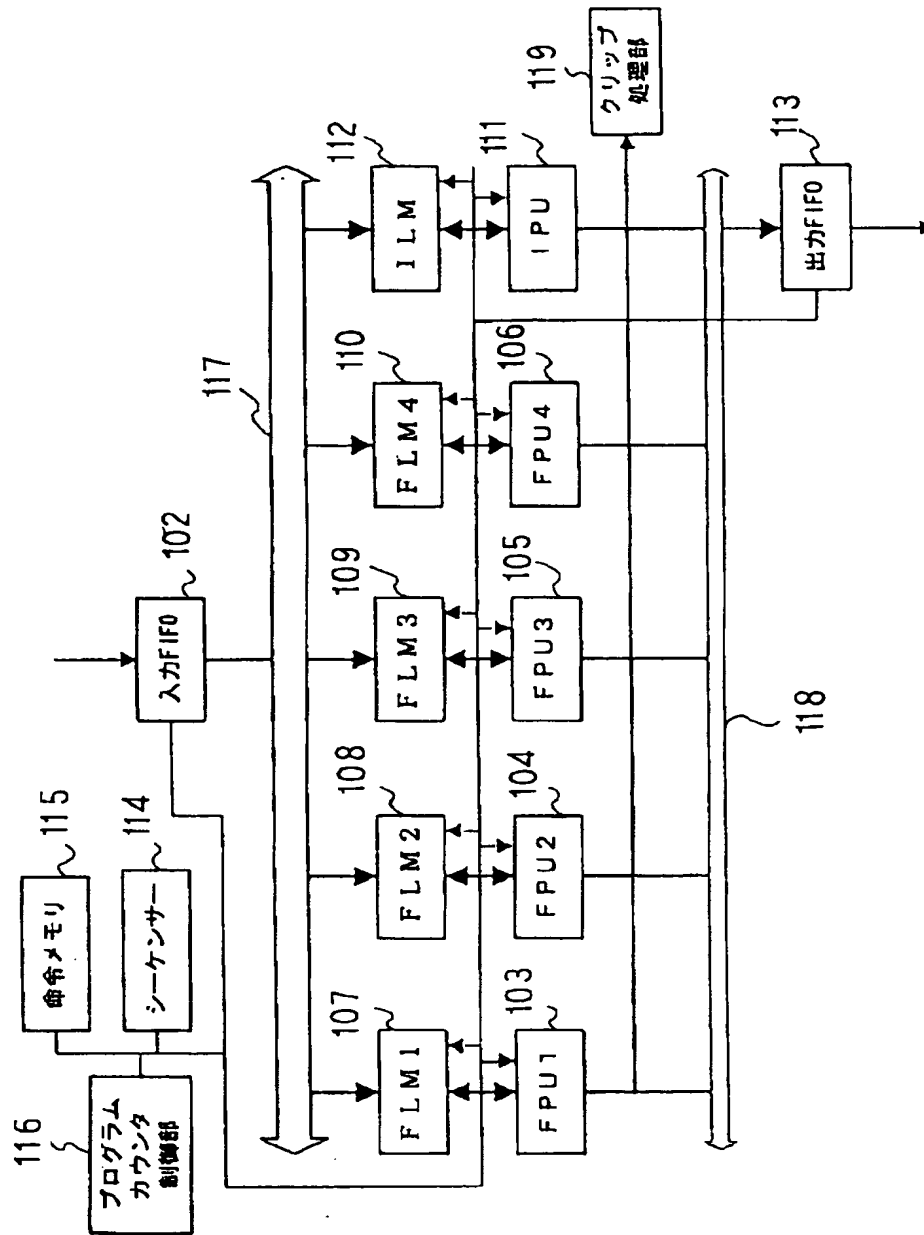
【図 4】



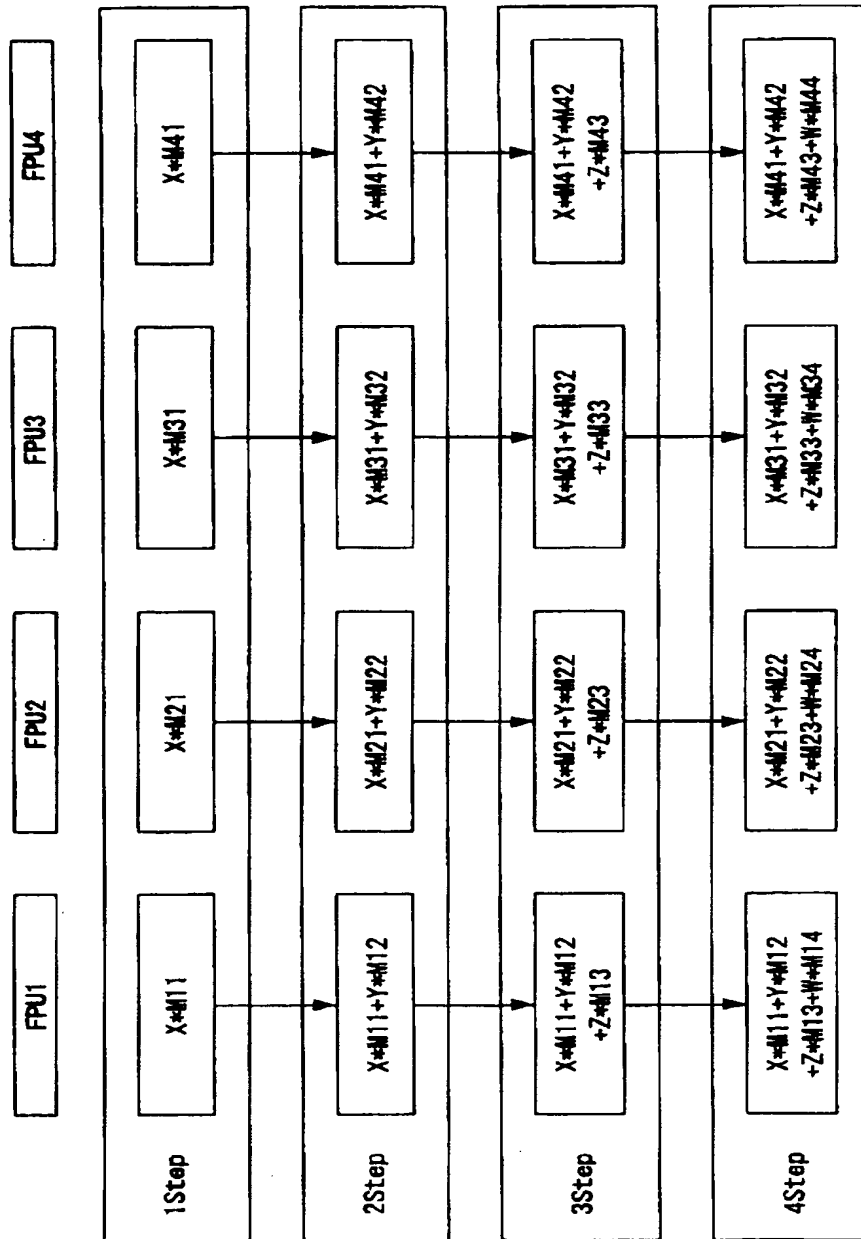
【図 23】



【図1】



【図 2】



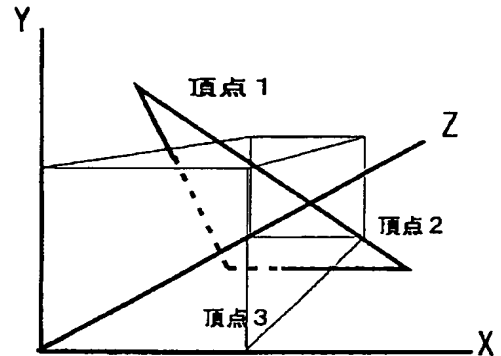
【図5】

(a)

Z値が $Z > Z_{MAX}$ の場合

XMIN		XMAX	
領域11	領域12	領域13	
010110	010100	010101	
<hr/>			YMAX
領域14	領域15	領域16	
010010	010000	010001	
<hr/>			YMIN
領域17	領域18	領域19	
011010	011000	011001	

【図6】



(b)

Z値が $Z_{MAX} > Z > Z_{MIN}$ の場合

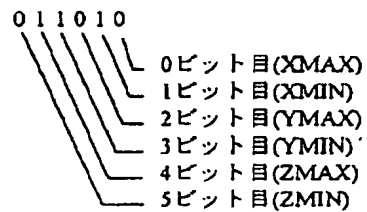
XMIN		XMAX	
000110	000100	000101	
<hr/>			YMAX
000010	000000	000001	
<hr/>			YMIN
001010	001000	001001	

(c)

Z値が $Z_{MIN} > Z$ の場合

XMIN		XMAX	
100110	100100	100101	
<hr/>			YMAX
100010	100000	100001	
<hr/>			YMIN
101010	101000	101001	

(d)

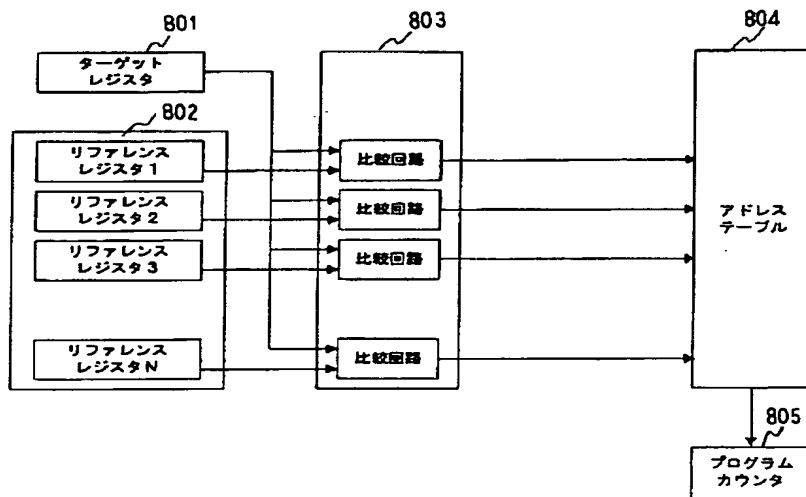


【図7】

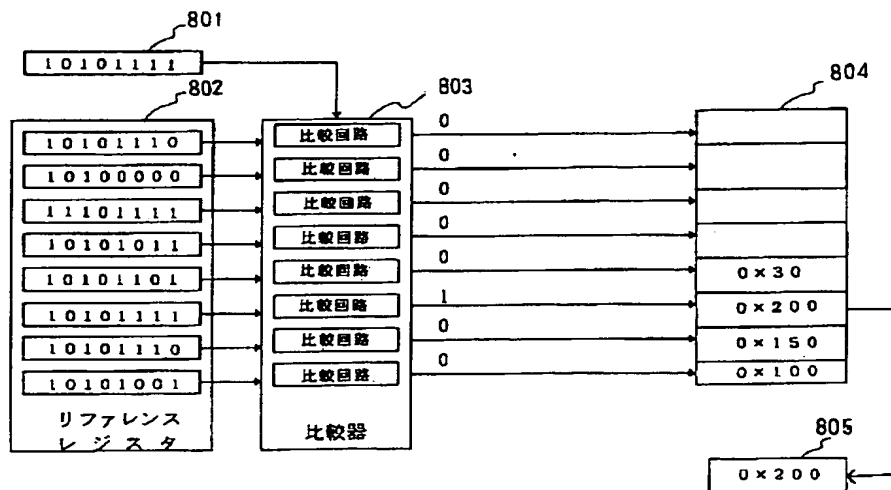
	クリップコード						クリップ 状態コード		
	シフト レジスタ1	シフト レジスタ2	シフト レジスタ3	シフト レジスタ1	シフト レジスタ2	シフト レジスタ3	ClipIn	ClipOut	Clip
ステップ1	00 00 00	10 00 00	00 00 00				0	0	1
ステップ2	10 00 00	00 10 00	00 00 00				0	0	1
ステップ3	00 10 00	00 00 10	00 00 00				0	0	1

第3フィールド
第2フィールド
第1フィールド

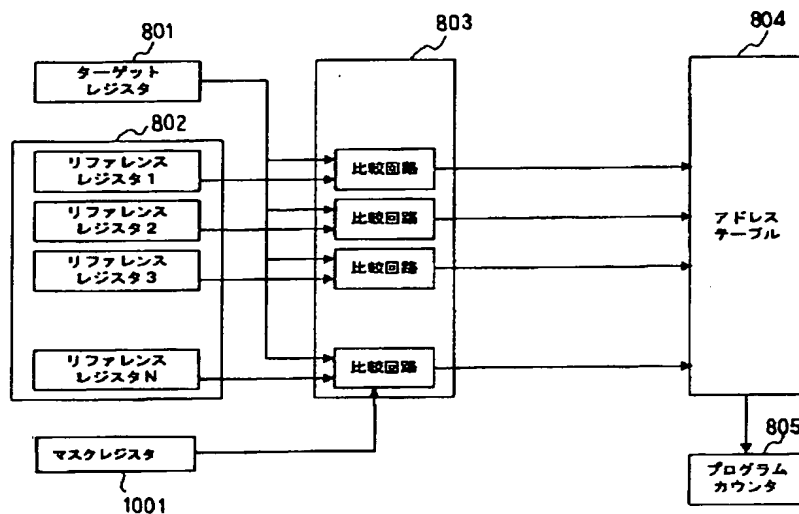
【図8】



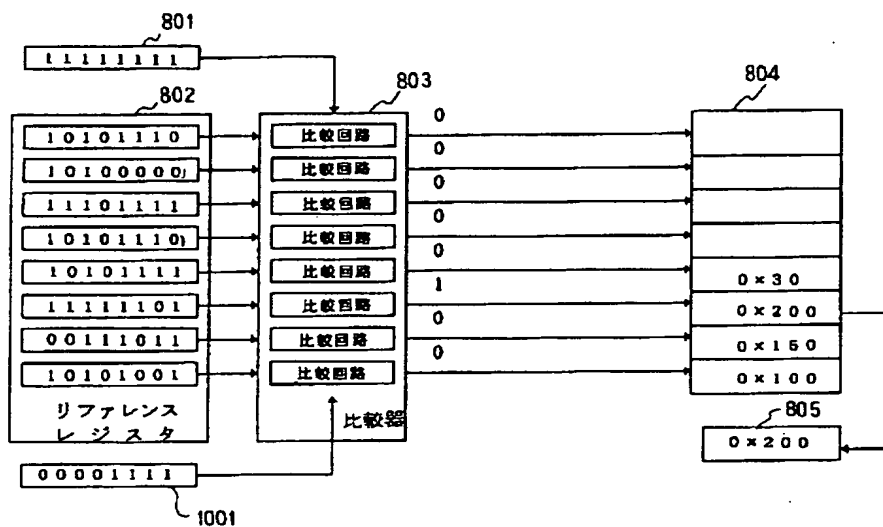
【図9】



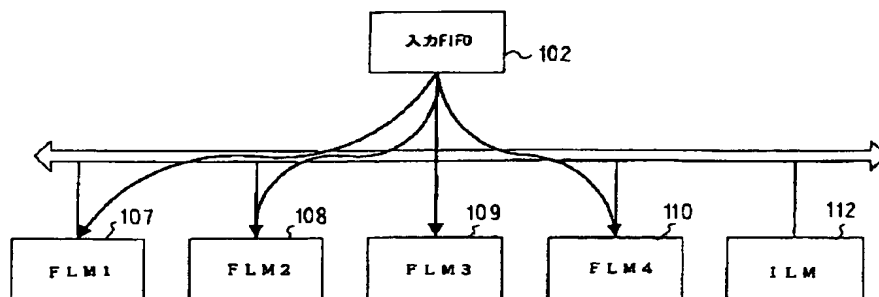
【図10】



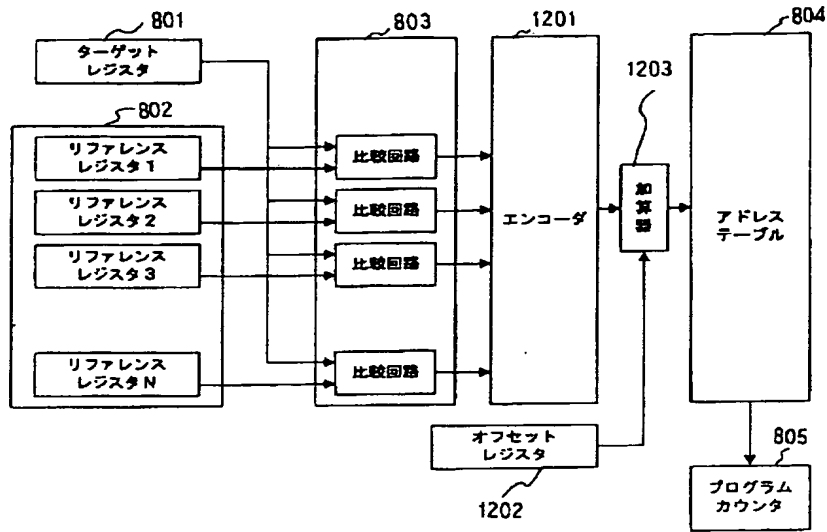
【図11】



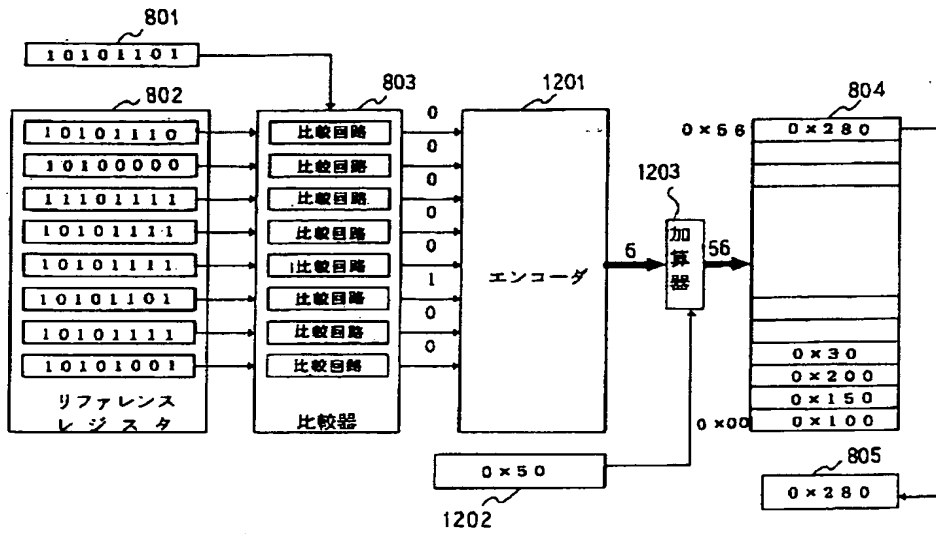
【図17】



【図12】



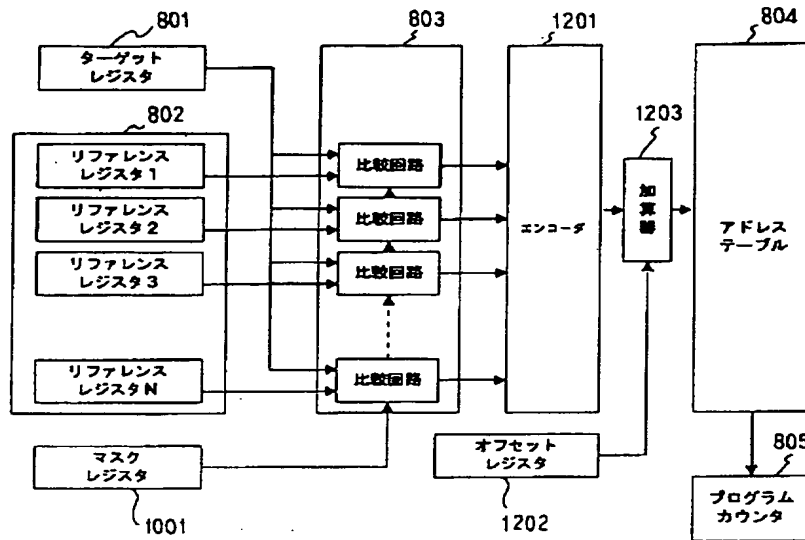
【図13】



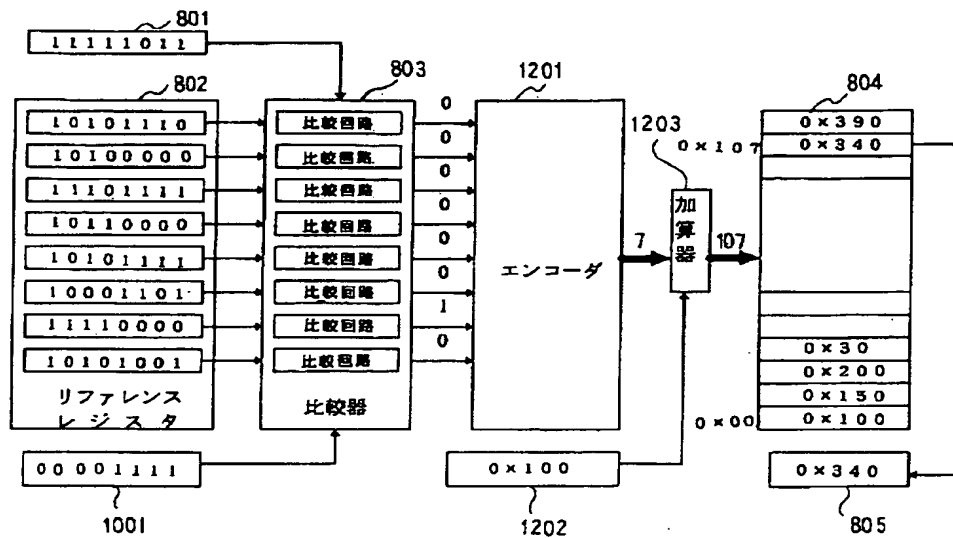
【図18】

コマンド	ソース フィールド	ソースアドレス	デスティネーション フィールド	デスティネーション アドレス
------	--------------	---------	--------------------	-------------------

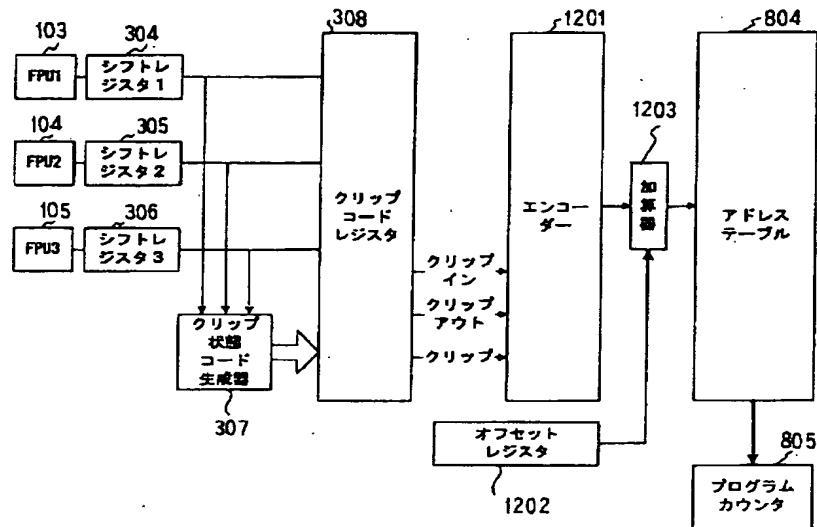
【図14】



【図15】



【図16】



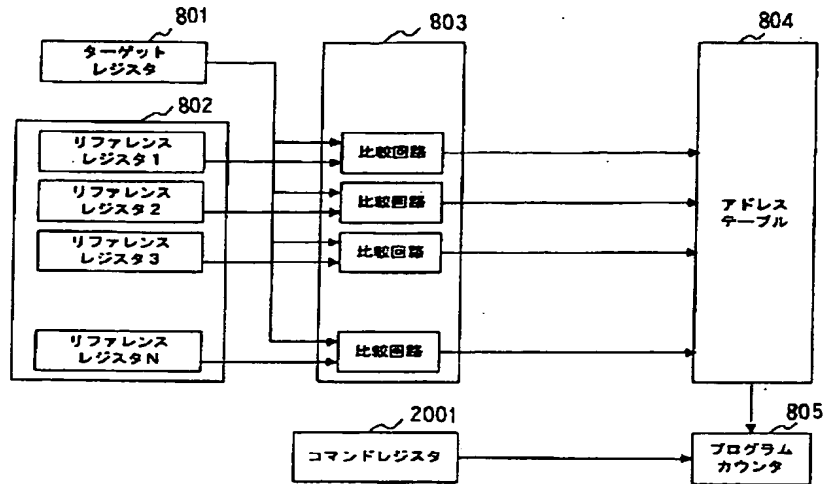
【図19】

フィールドコード	転送ソース/デスティネーション
000	内部レジスタ
001	ILM
011	マルチキャストFLM
100	FLM1
101	FLM2
110	FLM3
111	FLM4

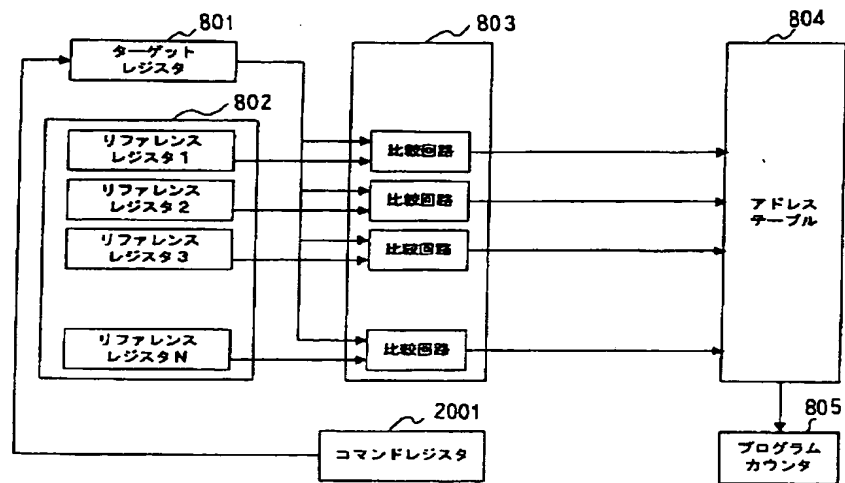
【図21】

コマンド 番号	プログラム アドレス	
------------	---------------	--

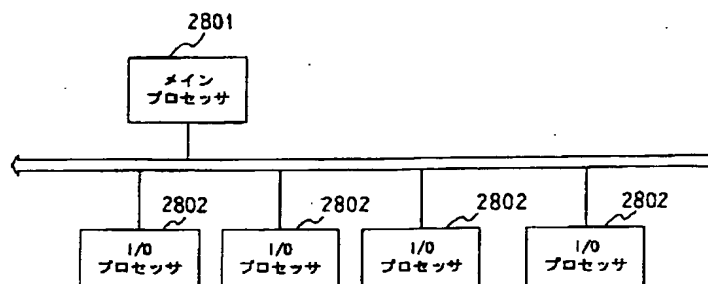
【図20】



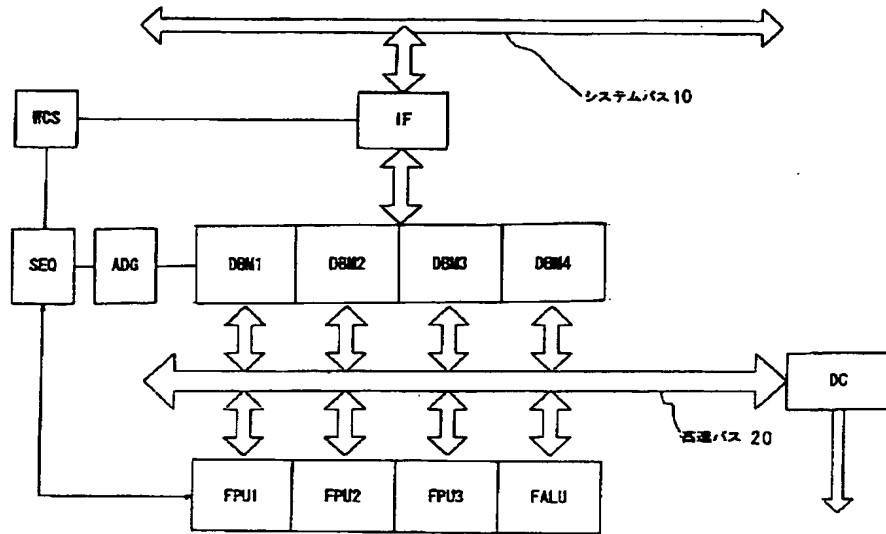
【図22】



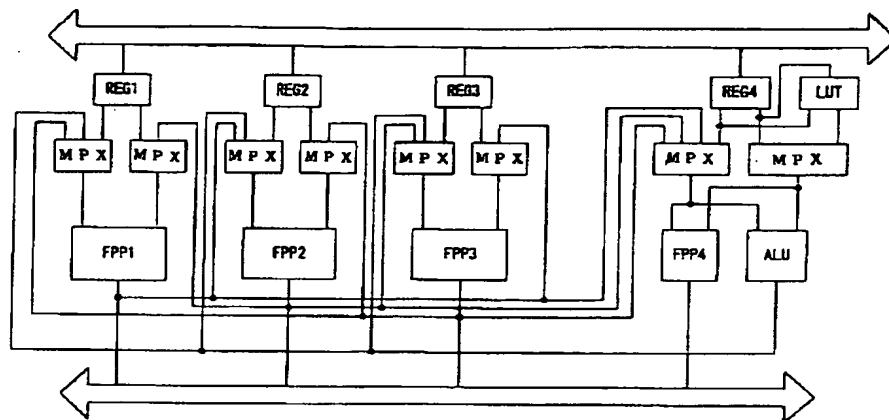
【図28】



【図 24】



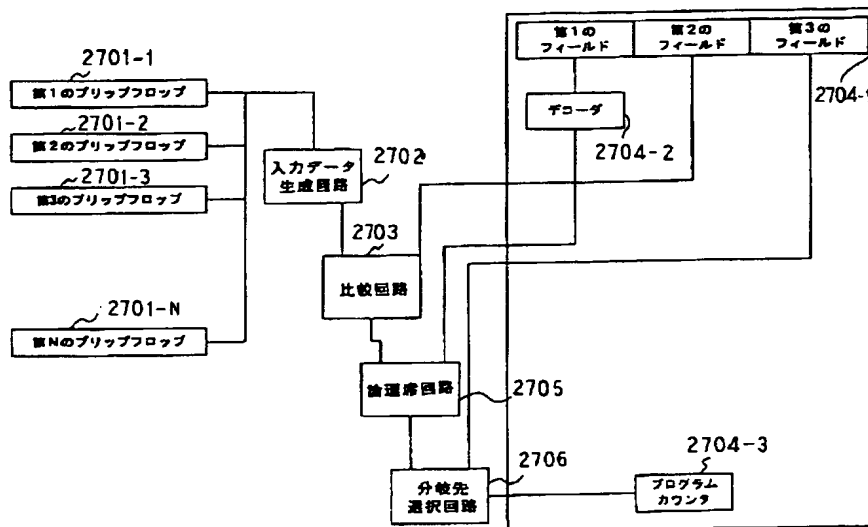
【図 25】



【図26】

No	FPP1	FPP2	FPP3	FPP4
1	$T11 \cdot X$	$T21 \cdot Y$	$T31 \cdot Z$	$T41 \cdot W$
2	$T11 \cdot X + T21 \cdot Y$		$T31 \cdot Z + T41 \cdot W$	
3	$T11 \cdot X + T21 \cdot Y +$ $T31 \cdot Z + T41 \cdot W$			
4	$T12 \cdot X$	$T22 \cdot Y$	$T32 \cdot Z$	$T42 \cdot W$
5	$T12 \cdot X + T22 \cdot Y$		$T32 \cdot Z + T42 \cdot W$	
6		$T12 \cdot X + T22 \cdot Y +$ $T32 \cdot Z + T42 \cdot W$		
7	$T13 \cdot X$	$T23 \cdot Y$	$T33 \cdot Z$	$T43 \cdot W$
8	$T13 \cdot X + T23 \cdot Y$		$T33 \cdot Z + T43 \cdot W$	
9			$T13 \cdot X + T23 \cdot Y +$ $T33 \cdot Z + T43 \cdot W$	
10	$T14 \cdot X$	$T24 \cdot Y$	$T34 \cdot Z$	$T44 \cdot W$
11	$T14 \cdot X + T24 \cdot Y$		$T34 \cdot Z + T44 \cdot W$	
12				$T14 \cdot X + T24 \cdot Y +$ $T34 \cdot Z + T44 \cdot W$

【図27】



フロントページの続き

(72)発明者 河合 浩行
 東京都千代田区丸の内二丁目2番3号 三
 菱電機株式会社内